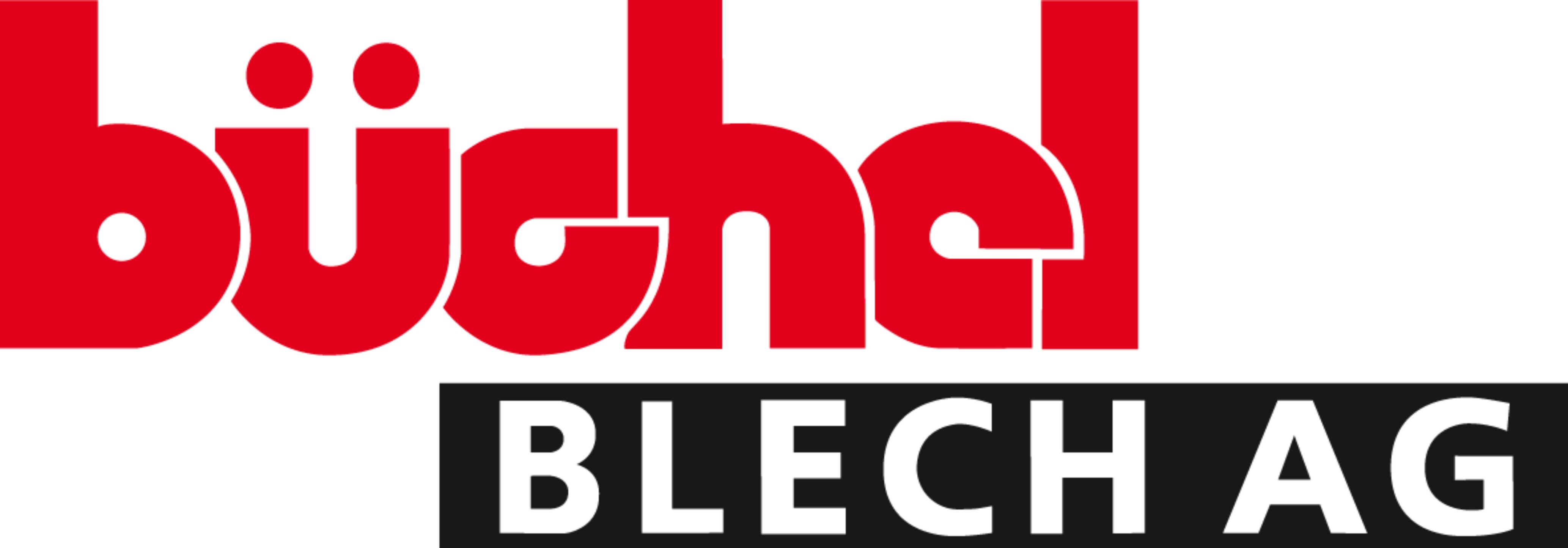


**KLAUS**

**DER LEUCHTTURMBAUER**





**Team 10, Chur**

* Erich Meyer
* Luzian Weber
* Michael Alig
* Romario Jäger
* Jeanot Zwick
* Mauro Stoffel



Inhalt:

[2 Kurzfassung 3](#_Toc516585759)

[3 Einleitung 3](#_Toc516585760)

[3.1 Teamvorstellung/Struktur 4](#_Toc516585761)

[3.2 Entwicklungsverlauf des Konzepts. 4](#_Toc516585762)

[4 Konzeptentscheid 5](#_Toc516585763)

[5 Teilfunktionen 7](#_Toc516585764)

[5.1 Übersicht 7](#_Toc516585765)

[5.2 Mechanik 8](#_Toc516585766)

[5.3 Fortbewegung 8](#_Toc516585767)

[5.3.1 Unterteil 8](#_Toc516585768)

[5.3.2 Oberteil 8](#_Toc516585769)

[5.3.3 Vorgehen 8](#_Toc516585770)

[5.3.4 Probleme/ Lösungen 9](#_Toc516585771)

[5.4 Elektronik 10](#_Toc516585772)

[5.4.1 Printplatte 10](#_Toc516585773)

[5.4.2 MPC 11](#_Toc516585774)

[5.4.3 Sensorik 11](#_Toc516585775)

[5.4.4 Encoder 12](#_Toc516585776)

[5.4.5 W-Lan 12](#_Toc516585777)

[5.4.6 Probleme und Lösungen 12](#_Toc516585778)

[5.5 Informatik 13](#_Toc516585779)

[5.5.1 Programmablauf 13](#_Toc516585780)

[5.5.2 Kommunikationskonzept 14](#_Toc516585781)

[5.5.3 Programmaufbau 15](#_Toc516585782)

[5.5.4 Probleme und Lösungen 16](#_Toc516585783)

[6 Schlussfolgerung 17](#_Toc516585784)

[6.1 Rückblick: 17](#_Toc516585785)

[6.2 Erfahrungen: 17](#_Toc516585786)

[6.3 Schlusswort 18](#_Toc516585787)

# Kurzfassung

Uns wurde die einmalige Gelegenheit geboten, in den ersten beiden Semestern des Studiums einen Roboter zu bauen. Aus allen Standorten der NTB wurden Teams gebildet, deren Auftrag es war einen Roboter zu bauen, um dann mit einem Partnerroboter aus einem anderen Team einen Turm zu bauen.

Die Bausteine des Turms sind Duplos, welche der Roboter aus den Spendern vorne auf dem Spielfeld entnimmt und dann aufeinanderstapelt, um die Basis eines Leuchtturms zu bilden.

Unserer Roboter heisst Klaus und besteht aus unterschiedlichen Systemen die miteinander interagieren.

# Einleitung

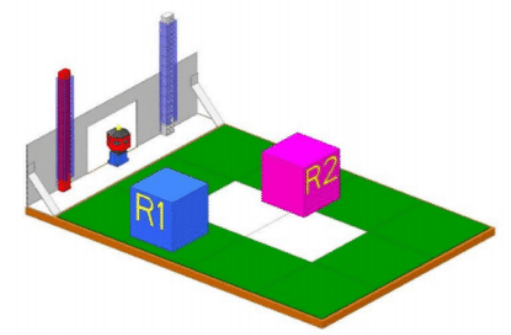
Als Team 10 aus Chur haben wir im Rahmen des Systemtechnikprojektes 2017/2018 die Aufgabe, einen Roboter zu entwickeln und mit Hilfe dessen eine interdisziplinäre, technische Problemstellung zu lösen. Es soll ein Leuchtturm aufgebaut werden, wozu 2 kooperierende Roboter genutzt werden. Team 2 und Team 3 in Buchs entwickeln je einen Partnerroboter. Das Zeitlimit, bis der Turm eigenständig in der geforderten Höhe und mit Leuchtturmspitze steht, beträgt drei Minuten. Zu erkennen ist die Dauer der zur Verfügung gestellten Zeit daran, dass die Turmspitze zu Beginn kurz aufblinkt und am Ende zu leuchten beginnt. Es muss jeweils eine Kommunikation zwischen den beiden Partnerrobotern stattfinden, um die Arbeitsschritte der autonom arbeitenden Roboter aufeinander abzustimmen. Jedem Team steht ein Budget in der Höhe von CHF 800.- zur Verfügung. Es dürfen jedoch Materialsponsoren angefragt werden. Das Systemtechnikprojekt unterliegt einem festgelegten Zeitfenster. Es müssen verschiedene Meilensteine eingehalten werden. Bis zum 26.06.2018 muss das Projekt endgültig abgeschlossen sein und einer Fachjury, sowie dem öffentlichen Publikum präsentiert werden. Um den Erfolg zu gewährleisten muss das Wort «Team» grossgeschrieben werden. Doch eine gute Zusammenarbeit gelingt nur, wenn jedes einzelne Teammitglied Eigenverantwortung und Eigeninitiative wahrnimmt.

Abb. 3‑1 Spielfeld

## Teamvorstellung/Struktur

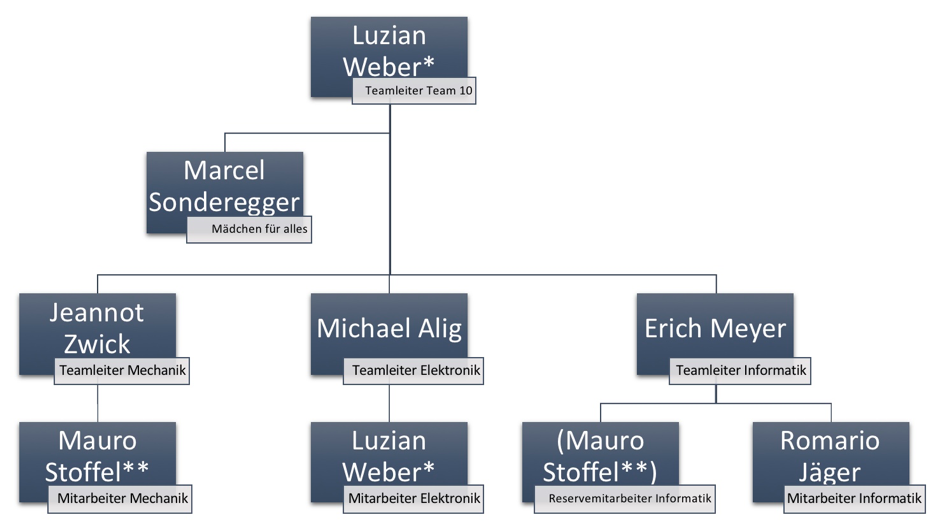


Abb. 3‑2 Organigramm

## Entwicklungsverlauf des Konzepts.

Der Ablauf bis zum Konzeptentscheid wurde in mehrere Teilschritte unterteilt:

* Ideensammlungsphase
* Skizzenphase (Bilder)
* Aufteilen der Teilaufgaben
* Beginn der Ideenanalyse auf Möglichkeiten und Durchführung
* Überarbeiten der Ideen -> neue Ideenfindungen
* Endgültige Prüfung der Kompatibilität mit Partnerteam
* Konzeptentscheid

# Konzeptentscheid

Für die Konzeptfindung wurde im Vorfeld ein Brainstorming durchgeführt und daraus ein morphologischer Kasten erstellt:



Abb. 4‑1 Morphologischer Kasten

Nach genauer Abschätzung aller Varianten und Gewichtung von dessen Vor- und Nachteile wurde ein Konzept festgelegt.

|  |  |
| --- | --- |
| Energieversorgung | Lithium-Polymer-Akkumulator |
| Kommunikation | Wifi Modul |
| Fortbewegung | 4 Räder, die zwischen den Noppen der Legoplatte fahren |
| Antrieb | Bürstenloser Gleichstrommotor mit Encoder und Servo-Motor |
| Greifmechanismus | Greifer mit Magnet |
| Turm bauen | «Gabelstapler», ausfahrbarer Arm |
| Orientierung | IR-Sensoren und Encoder |

Für die Entscheidung des Konzeptes war die Zusammenarbeit mit den beiden Partnerteams von Buchs, Team 2 und 3, von grosser Bedeutung. Nach intensiven Diskussionen wurde eine endgültige Entscheidung gefällt, die folgenden Inhaltspunkte behält:

-Team 10 operiert auf der linken Spielfeldhälfte, Team 2 /3 auf der rechten.

-Team 10 legt den ersten Stein.

-Danach werden abwechselnd Steine gelegt, bis die erforderte Höhe erreicht ist.

-Team 2/3 setzt die Turmspitze.

-Die Hilfssteine werden wie folgt eingesetzt:

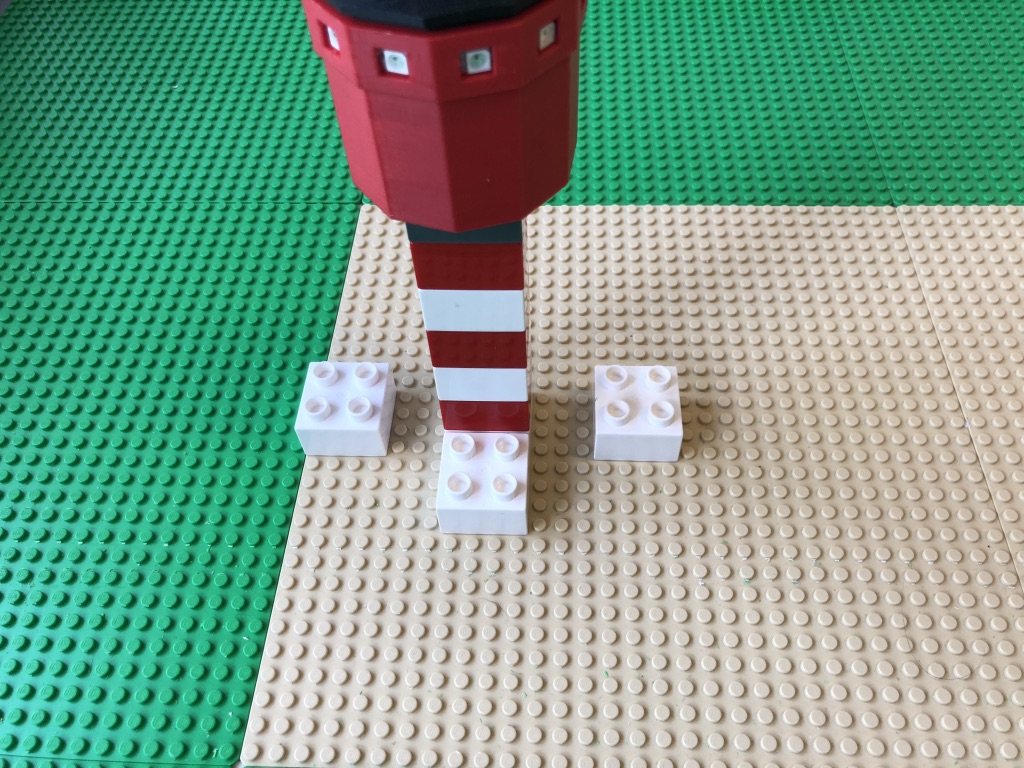


Abb. 4‑2 Hilfssteine: Die drei weissen Steine neben und vor dem Leuchtturm

# Teilfunktionen

## Übersicht

|  |  |  |
| --- | --- | --- |
| **Startposition**  Je ein Roboter startet auf einer Seite des Spielfeldes. Die Entfernung zum Legostein-Spender ist willkürlich. Jedoch darf die Platzierung in der Breite von den Teams frei gewählt werden. Die drei frei verwendbaren Legosteine werden als Anschläge beim Leuchtturmstandort und als Distanzhilfe beim Rückwärtsfahren verwendet.  Wenn das Startsignal gesendet wurde, fahren beide Roboter nach vorne um sich den ersten Duplostein zu holen. | **Position Vorne**  Durch einen Distanzsensor merkt Klaus, wenn er vorne ist. Wenn er den Stein mit dem Magnetgreifer hat, fährt er zurück auf die Höhe der Leuchtturm Baustelle.  Beim Zurückfahren erkennt ein weiterer Distanzsensor den frei platzieren Legostein und Klaus hält an. Wenn der Partnerroboter das Signal schickt, dass die Baustelle frei ist dreht sich Klaus. | **Bau-Position**  Nach der 90 Grad Drehung setzt Klaus den transportierten Stein auf der Baustelle fest. Anschliessend hebt er seinen Arm wieder, Dreht zurück und sendet dem Partnerroboter ein Signal, dass die Baustelle bereit ist.  Der ganze Ablauf wird nun solange wiederholt, bis Klaus den neunten Duplostein gesetzt hat. Danach setzt unser Partnerroboter noch die Spitzte das Leuchtturms auf und die Aufgabe ist erledigt. |

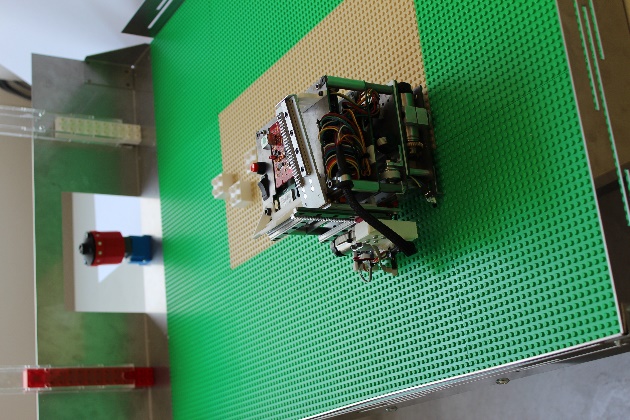


Abb. 5‑1 Startposition

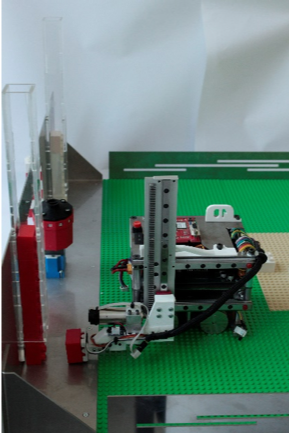


Abb. 5‑2 Position Lego greifen

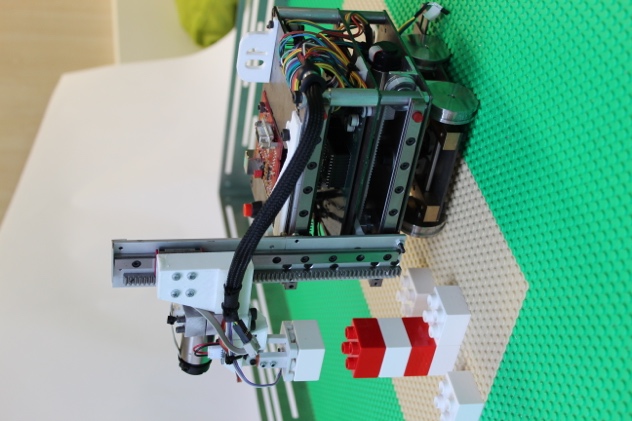


Abb. 5‑3 Bauposition

## Mechanik

## Fortbewegung

### Unterteil

#### Bodenplatte

Auf der Bodenplatte des Fahrwerks wurden der Motor für die Fortbewegung, die Achsenlager für die Räder, sowie der Drehstützen für die Verbindung zwischen Fahrwerk und Oberteil befestigt. Ausserdem wurde an der Bodenplatte ein Lichtsensor befestigt, welcher sicherstellt, dass der Roboter vor dem Ablegen des Steines am richtigen Ort ist.

#### Deckplatte

Auf der Deckplatte des Fahrwerks wurde eine gut gleitende Platte aufgebracht, um eine reibungsarme und ebene Oberfläche für die Drehung des Oberteils zu schaffen.

### Oberteil

#### Drehplatte

Auf der Drehplatte des Oberteils befinden sich der Antriebsmotor für die Drehbewegung, der Antriebsmotor für die horizontale Bewegung des Armes, welche über einen Keilriemen auf den Arm übertragen wird, sowie das Gegenstück zur Drehstütze.

#### Zwischenplatte

Auf der Zwischenplatte werden die elektronik-Platinen befestigt.

#### Deckplatte

Auf der Deckplatte des Oberteils wird die Rampe befestigt, welche für das Aufrichten des Klapparms zuständig ist. Ausserdem ermöglichen Aussparungen in der Deckplatte einen erleichterten Zugang auf die darunterliegende Elektronik.

#### Greifarm

Der Arm wird an einer vertikalen Gleitschiene befestigt, welche wiederum an zwei horizontalen Gleitschienen befestigt sind, welche an der Boden- und Deckplatte des Oberteils angebracht werden. Er besteht aus dem Arm und dem Greifer. Der Arm ist das Verbindungsstück zwischen Schiene und einem Servo-Motor. Dieser erlaubt es, den an ihm angebrachten Greifer um 90° zu drehen.

### Vorgehen

Wir wussten seit der Spezialisten Woche, dass der Roboter eine Kraft von ungefähr 45 Newton überwinden muss, um den ersten Stein auf das Spielfeld zu drücken. Daher war es uns wichtig, dass der Roboter genug schwer ist, so dass er nicht wegkippt, wenn er versucht den Stein anzudrücken. Daher haben wir bei allen fünf Platten Stahl verwendet, um das nötige Gewicht zu erreichen.

Bei der Dimensionierung der Fortbewegungsmotoren kamen folgende Berechnungen zustande:



Bei der Dimensionierung der anderen Motoren haben wir uns mit dem Informatikteam und dem Elektronikteam besprochen und ausserdem Rat von einem Mitarbeiter von Faulhaber eingeholt.

Dies ergab die folgenden Motoren und Getriebe von Faulhaber:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Anwendung | Motor | Drehmoment | U/min | Getriebe |
| Antrieb | 2232R012SR | 10 mNm | 5510 min-1 | 173:1 |
| Drehung | 2232R012SR | 10 mNm | 5510 min-1 | 173:1 |
| Horizontale Verschiebung | 1524T012SR | 2.9 mNm | 4130 min-1 | 485:1 |
| Vertikale Verschiebung | 2232R012SR | 10 mNm | 5510 min-1 | 308:1 |

### Probleme/ Lösungen

Eines unserer grössten Probleme von Anfang an war, dass uns entscheidende Erfahrungen im Umgang mit dem CAD Programm sowie dem Konstruieren selbst fehlten. Wir stellten schnell fest, dass wir möglichst wenig Teile aus Kunststoff herstellen durften und stattdessen Aluminium oder Stahl verwenden mussten. Dadurch erhöht sich die Stabilität und das Risiko eines Materialversagens wird minimiert. Nach dem Meilenstein M3 wurde uns bewusst wir sehr wir mit der Mechanik im Rückstand waren und mussten daher handeln. Wir überarbeiteten unsere Problemstellen nochmals und vereinfachten sie soweit wie möglich. Wir versuchten möglichst viele Teile zu bestellen anstatt selbst zu konstruieren. Als uns klar wurde, dass wir eine zu tiefe Übersetzung der Getriebe bestellt hatten bestellten wir neue, wobei Faulhaber so grosszügig war uns diese zur Verfügung zu stellen. Doch diese waren um einiges länger als die Vorherigen. Dies stellte uns erneut vor Probleme auf Grund von Platzmangel. Als wir dann noch ungeeignete Zahnräder bestellt hatten die keine Befestigungsmöglichkeit boten, entschieden wir uns dafür als erstes darauf zu achten, dass die Funktionstüchtigkeit des Roboter gewährleistet ist und uns erst dann um die Abmasse zu kümmern. Da die Zeit drängte haben wir die Teile bei einem Teammitglied zuhause gefertigt, es hätte zu viel Zeit in Anspruch genommen für jedes Detail auf Buchs zu fahren und mit dieser Werkstatt zusammen zu arbeiten, oder diese neu zu bestellen.

Im Nachhinein haben wir gelernt, dass man den gesamten Roboter inklusive Motoren, Schrauben, Zahnräder etc. im CAD simulieren sollte um vorgängig auszuschliessen, dass irgendwo ein Platzmangel auftaucht an welchem sich Teile schneiden. Zudem sollten extern gefertigte Teile möglichst früh bestellt werden aufgrund von langen warte Fristen.

## Elektronik

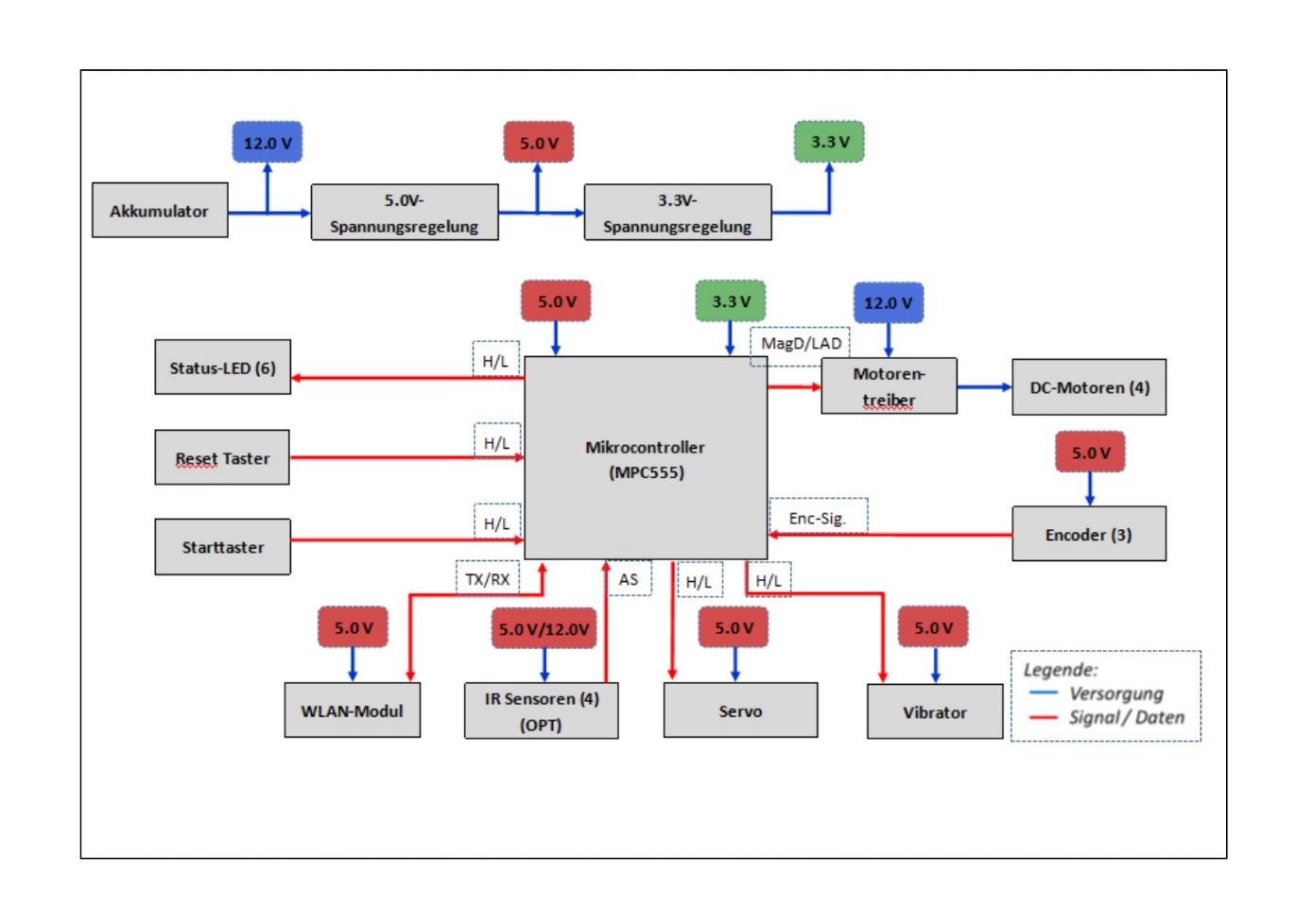


Abb. 5‑4 Blockschaltbild

Das Blockschaltbild ist eine übersichtliche Darstellung für die Aufteilung der Elektronik. In Realität wurden zwei separate Printplatten erstellt.

### Printplatte

Abb. 5‑5 3D Ansicht untere Printplatte

Die untere Printplatte beinhaltet:

* Spannungsversorgung 12V
* Transformation von 12V zu 5V
* Transformation von 5V zu 3.3V
* 2 Motorentreiber Locked Antiphase
* 1 Motorentreiber Sign Magnitude
* Signalisations-LED für Spannungsversorgung

Die Aufgabe der Spannungsversorgungsversorgung bzw. des Motorentreibers besteht darin, die gesamte Elektronik mit Energie zu versorgen sowie die verbauten Motoren bei Bedarf anzusteuern.

Als Energiespeicher dient ein Lithium-Polymer-Akkumulator (11.1 V / 1300 mAh). Dieser weist bei geringer Baugrösse

eine hohe Energiedichte auf. Auf dem Print wurden diverse Stützkondensatoren positioniert, um allfällige Spannungsschwankungen zu eliminieren. Somit kann gewährleistet werden, dass die Elektronik, auch bei hoher Stromaufnahme, mit einer konstanten Spannung versorgt wird.

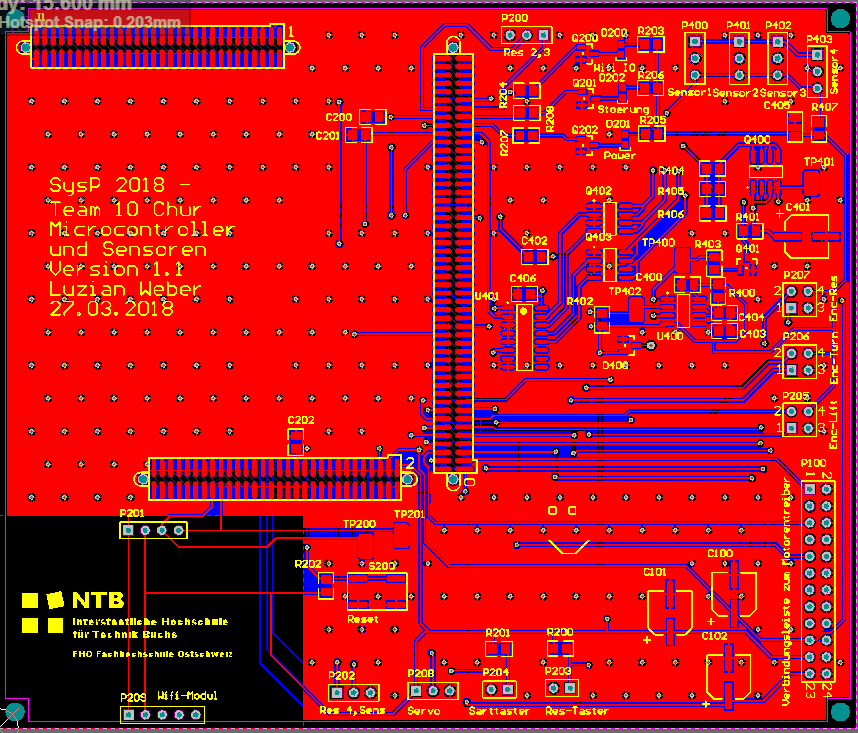
Der Drehmechanismus wird über einen Schneckenantrieb angetrieben. Dieses Antriebskonzept weist eine hohe Selbsthemmung auf. Aufgrund dessen benötigt dieser Motor, wie auch der des Vibrationsmotors, kein Haltemoment und wird deshalb mit dem Sign Magnitude-Signal betrieben. Alle anderen Motoren werden mit einer entsprechenden Zahnradpaarung mit der anzutreibenden Komponente verbunden. Diese Motoren benötigen ein Haltemoment und werden deshalb mit dem Locked Antiphase-Signal angesteuert.

Abb. 5‑6 PCB obere Printplatte

Somit kann gewährleistet werden, dass sich diese Motoren, auch im inaktiven Zustand nicht nur schwer drehen lassen.

Die obere Printplatte ist mit folgenden Bauteilen bestückt:

* IR-Sensore
* Start- und Reservetaster
* Vibrator Anschluss
* Wifi-Modul
* Encoder Anschlüsse
* Signalisations-LED
* Anschlussplatte MPC 555
* Resett-Taster

### MPC

Der MPC 555 ist das Herzstück und das Gehirn des Roboters. Von ihm aus werden sämtliche Ansteuerungen für Motoren, Sensoren, LEDs, etc. getätigt.

### Sensorik

Für die Orientierung des Roboters wurden IR-Sensoren eingesetzt. Der IR-Emitter sendet IR Licht aus, das an einem Objekt reflektiert. Anhand der Intensität des IR Lichtes, welches beim IR Empfängers (Receiver) ankommt, kann auf die Distanz zwischen dem reflektierenden Objekt und IR Sensor geschlossen werden.

Verwendet wurden die HLC 1395 Sensoren, welche ideal für kurze Distanzen eingesetzt werden können. Eine genauere Beschreibung kann dem Datenblatt auf der CD entnommen werden.

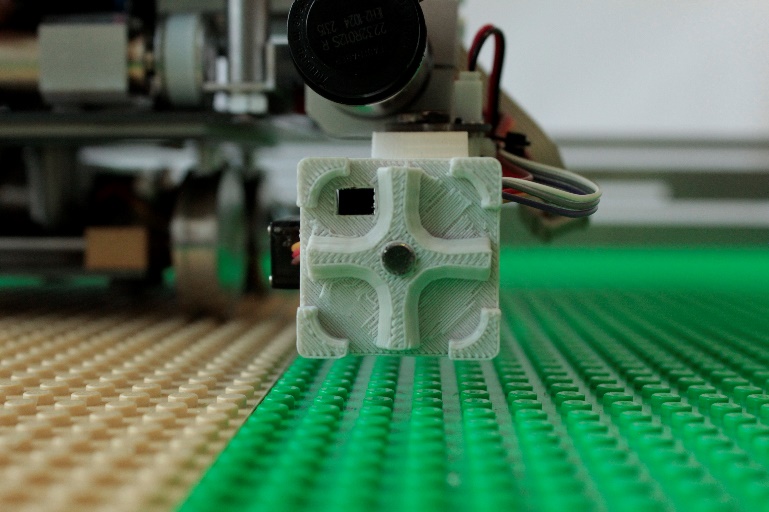


Abb. 5‑7 Sensor vorne (im Greifer)

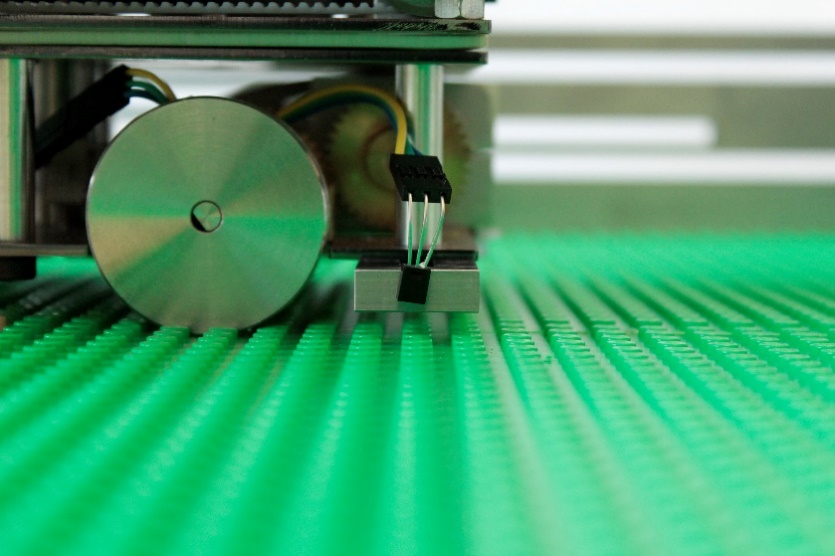


Abb. 5‑8 Sensor hinten

Verwendete Bereiche der Sensoren:

* In Greifer für die Vorwärtsfahrt
* Unten am Roboter für die Rückfahrt

### Encoder

Für die genaue Positionsmessung wurden Encoder eingesetzt. Die Encoder der Serie IEH2-4096 wurden direkt von Faulhaber mit den Motoren mitgeliefert. Eine genauere Beschreibung kann dem Datenblatt auf der CD entnommen werden.

Einsatzbereich der Encoder:

* Positionsmessung des Armes horizontal
* Positionsmessung des Armes vertikal
* Drehmotor

### W-Lan

Für die Kommunikation wurde auf die empfohlene Variante der NTB gesetzt, das WLAN Modul. Das Wifi Modul RN-131GXS ist ein fertig bestücktes Bauteil, das von der NTB zu Verfügung gestellt wurde. Für eine optimale Kommunikationsverbindung mit dem Partnerteam wurde das Modul frei von metallischen Umhüllungen platziert. Auch das Polygon auf unserer Printplatte wurde unter dem Modul ausgeschlossen. (Siehe Bild PCB)

### Probleme und Lösungen

Anfänglich waren wir etwas überfordert mit dem in der Spezialistenwoche erlernten Programm. Weder Michael noch Luzian hatten Erfahrungen mit Computerprogrammen für Leiterplatten. Da im Standort Chur keine Experten mit Erfahrung in Altium Design vorhanden waren musste das Team Elektronik immer mit Buchs korrespondieren um Hilfe einzuholen.

Nach der Bestellung der PCB’s erfuhren wir das ein Spannungsregler nicht mehr lieferbar war und wir mussten eine Alternative suchen. Der Ersatzregler hatte leider nicht mehr die gleichen Abmasse und wir mussten beim Bestücken die Verbindungen mit Drähten zusammenlöten.

Dass die Vorwiderstände der IR-Sensoren falsche Werte hatten bemerkten wir beim Austesten und wechselten diese aus.

## Informatik

Kleine Einführung in die Funktion der Informatik

### Programmablauf

Abb. 5‑9 Ablaufdiagramm

Der Roboter kennt insgesamt 10 verschiedene Zustände.

Im Ersten wird die Kommunikation mit dem Partnerroboter aufgebaut und alle Motorentreiber unter Strom gesetzt. Anschliessend wird das Startsignal gesendet und der nächste Zustand wird aktiviert.

Hier fährt Klaus das erste Mal nach vorne und fährt seinen Greifarm aus. Wenn der Sensor im Greifer vorne erkennt das er vorne ist, stoppt der Roboter.

Jetzt starte er einen Timer damit er weiss, wann der den Arm heben darf. Dann fährt er zurück und sobald der Timer abgelaufen ist, fährt auch der Greifarm nach oben.

Wenn nun der hintere Sensor den Legostein auf der linken Seite erkennt (welchen wir vor dem Start dort setzten dürfen), hält der Roboter an und wartet bis das Signal vom Partnerroboter kommt, dass die Baustelle frei ist. Weil wir den ersten Stein setzten, wird beim ersten Mal nicht darauf gewartet und Klaus drehen direkt ab.

Ist der Legostein nun richtig positioniert, fährt der Arm nach unten und drückt den Stein an. Hiernach fährt der Arm wieder nach oben und Klaus dreht sich wieder Richtung Legospender. Gleichzeitig sendet Klaus dem Partnerroboter ein Signal mit der Anzahl gesetzter Legosteinen, was gleichzeitig bedeutet, dass nun er seinen Stein setzten kann.

Ist nun die gewünschte Höhe von neun Legosteinen erreicht. Stoppen wir und senden das erste Stoppsignal. Wenn dies nicht der Fall ist, Fährt der Roboter wieder nach vorne und holt den nächsten Stein, bis die Höhe erreicht ist.

### Kommunikationskonzept

Abb. 5‑10 Sequenzdiagramm

Die Kommunikation wird über Fire&Forget-System gemacht, es wird keine Bestätigung für das erhalten einer Nachricht verschickt. Die einzige Ausnahme dazu ist das Signal 222, das zu Beginn verschickt wird, um zu überprüfen, ob der Partnerroboter bereit ist und die Kommunikation steht. Falls dies nicht der Fall ist, wird das Signal nach kurzer Zeit wiederholt. Sobald die Bestätigung kommt, wird das Startsignal an den Leuchtturm und den Partnerroboter geschickt.

Während des Bauens wird jeweils das Setzen eines Steins mit einer Nachricht an den Partnerroboter bestätigt, um Komplikationen mit der Nummer 10, die beim Testen Probleme verursachte, zu vermeiden, wird jeweils die Höhe in Steinen+100 gesendet. Sobald unser Roboter den letzten Stein gesetzt hat, schickt er das Schlusssignal an den Leuchtturm.

### Programmaufbau

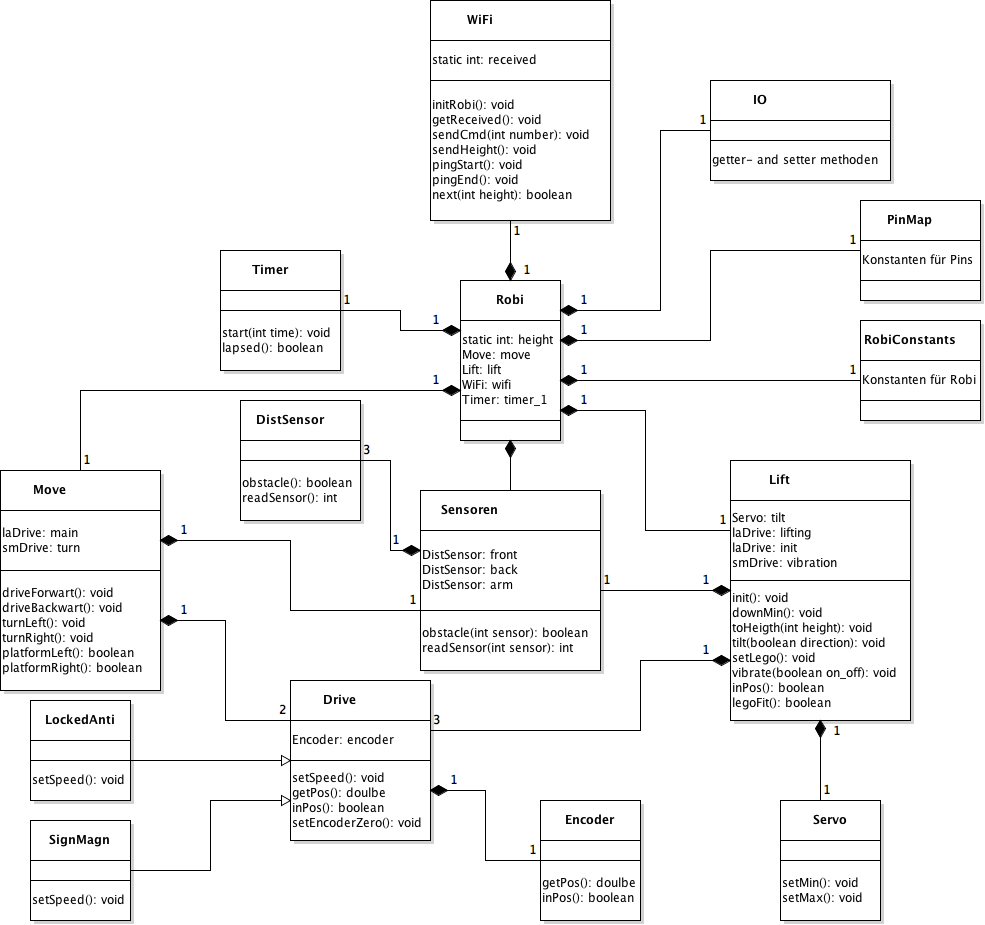


Abb. 5‑11 Klassendiagramm

Die Klasse Robi ist die Klasse, von der aus der Ablauf gesteuert wird. Alle Funktionen anderer Klassen werden von dieser Klasse aus gesteuert. Die verschiedenen Zustände sind im Abschnitt Zustandsdiagramm genauer erläutert.

Die Klasse Timer beinhaltet eine Stoppuhr, durch welche man Zeitabstände, die nicht durch Tasks gesteuert werden, messen kann.

Die Kommunikation wird durch die Klasse Wifi gesteuert. Der genaue Ablauf der Kommunikation sind im Abschnitt Kommunikationsdiagramm erklärt.

Die Klasse IO instanziiert Ein- und Ausgänge, in der Klasse PinMap sind alle benutzten Pins aufgeführt. Diese Klassen helfen, Ordnung zu halten, da man immer weiss, wo ein Pin zu finden ist und man mit einem Klick den Wert für alle Aufrufe dieses Pins verändern kann. In RobiConstants werden Konstanten gespeichert, um diese mit einem Klick verändern zu können.

Sensoren beinhaltet eine Liste der DistSensoren, diese beiden Klassen verwalten die Distanzsensoren und beinhalten Methoden um Hindernisse zu erkennen oder zu testen, ob ein bestimmter Grenzwert erreicht wurde.

Servo steuert den Servomotor, für dieses Projekt wurden nur die Maximal- und Minimalposition gebraucht.

Die Klasse Encoder speichert die Position eines Motors. Ein Encoder kann mit einem Faktor initialisiert werden, damit alle Übersetzungen unterstütz werden.

Drive ist eine abstrakte Klasse, die entweder vom Typ LockedAnti (=Locked Antiphase) oder SignMagn (=Sign Magnitude) ist. Die verschiedenen Ausprägungen müssen die Methode setSpeed überschreiben, damit man beide Arten von Motoren gleich ansteuern kann. Ausserdem wird beim Instanziieren eines Motors festgelegt, ob dieser über einen Encoder oder über Sensoren gesteuert wird.

Die Klasse Move verwaltet die Bewegung des Roboters und steuert zwei Sensoren sowie ein Locked-Antiphase-Motor ohne Encoder für die Bewegung auf dem Spielfeld und ein Sign-Magnitude-Motor mit Encoder für die Drehung des Roboters.

Die Klasse Lift ist ähnlich wie die Klasse Move, steuert aber den Greifmechanismus.

Beide Klassen stellen Methoden für die Klasse Robi zur Verfügung, um diese Klasse möglichst übersichtlich zu halten.

### Probleme und Lösungen

Die erste Version des Programmes hatte nur wenige Klassen, was das Testen und die Fehlersuche sehr schwer gestalten würde. Ausserdem sind die einzelnen Klassen so sehr unübersichtlich. Auf einen Ratschlag des damaligen Betreuers, Ulrich Hauser, wurde das Programm neu strukturiert. Die oben genannten Diagramme basieren auf diesem Stand.

Von den Studenten, die das Projekt bereits hinter sich hatten, wurde geraten, Klassen wie PinMap einzubauen, um Konstanten an einem Ort zu zentrieren, was Änderungen vereinfacht.

Während des Testens gab es ein Problem mit den Locked Antiphase Motoren. Das Problem war, dass die Initialisierung an der falschen Stelle im Programm stattfand.

Zu Beginn machte das WLAN-Modul Probleme. Diese sind auf Konflikte mit der Task-Superklasse zurückzuführen. Als diese Probleme gelöst waren, funktionierte es immer noch nicht, auf dem Testboard funktionierte es aber. Somit wurde das WLAN-Modul ausgetauscht, mit dem Neuen funktioniert es.

# Schlussfolgerung

## Rückblick:

Vor gut 10 Monaten starteten wir das Studium Systemtechnik. Berühmt, berüchtigt war uns schon im Voraus das Systemtechnikprojekt mit dem Bau eines Roboters von älteren Studenten bekannt. Uns wurde aber auch mitgeteilt, dass es kein anderes Projekt während dem ganzen Studium gibt, wo der Lerneffekt grösser sei, das Team mehr zusammenschweisst und die fächerübergreifenden Disziplinen besser zusammenspielen. Demensprechend stiegen wir motiviert in das Projekt ein.

Nach der Konzeptfindung im Meilenstein 1 ging es darum die Ideen umzusetzen. Mit viel Fleiss und Einsatz probierten wir unsere ungünstigen beruflichen Vorbildungen zu kompensieren, was uns manchmal besser, manchmal weniger gut gelang.

Dank der guten Betreuung in der Elektronik von Experten aus Chur konnte der gelernte Stromer Luzian und der Automech Michael zwei funktionierende Leiterplatten herstellen. Einzig ein Widerstand musste ausgewechselt und der Anschluss der Spannungsversorgung überarbeitet werden.

Im Team Mechanik probierte der gelernte Schlosser so gut es ging mit dem neu erlernten Programm Creo CAD Zeichnungen erstellen. Nachdem Laura das Studium nach dem ersten Semester abgebrochen hatte wechselte der Maturand Mauro zum Team Mechanik um Jeannot zu unterstützen. Jedoch hatte auch er keine Creo Erfahrungen. Dank seinem privaten 3D Drucker konnte er unzählige Teile für den Roboter drucken und leistete somit einen wichtigen Beitrag zur Realisierung des Projekts. Einige mechanische Teile die fehlerhaft waren konnten in Michaels Werkstatt auf der Drehbank passend gemacht werden.

Das Informatikteam, das nach Mauros Wechsel nur noch aus einem Automatiker und einem Matura Absolvent bestand war ebenfalls personell limitiert. Da beide jedoch auch in der Freizeit begeisterte Programmierer sind, konnten sie alle Probleme mit Bravour meistern.

Wir sind stolz darauf, dass wir aus allen Situationen immer eine Lösung fanden und mit grosser Teamarbeit und gegenseitiger Unterstützung bestehend aus nur sechs Studenten einen funktionierenden Roboter bauten.

## Erfahrungen:

Mit diesem Projekt konnten wir viele Erfahrungen sammeln, einerseits in technischen Bereichen wie der Vorgehensweise bei der Konzeptfindung und der Fehlerbehandlung, aber auch im Umgang mit Mitmenschen, beispielsweise dem Einbeziehen von externen Personen und im Umgang mit Kritik.

Das Bauen eines Roboters ist die eine Sache, das Präsentieren der Ideen und Meilensteine dem «Kunden» oder wie in diesem Projekt den Experten eine andere Sache. Somit konnten wir auch viel in Sachen Präsentationstechnik für unsere Berufslaufbahn mitnehmen.

Sich Hilfe zu suchen und die richtigen Experten beizuholen war ein wichtiger Eckpfeiler für die erfolgreiche Umsetzung. Wir haben gelernt, dass man in einer überforderten Situation immer jemand finden kann, der Lösungen hat.

## Schlusswort

Das Systemtechnikprojekt hatte es definitiv in sich und alle Erwartungen erfüllt. Es war eine sehr intensive Zeit, die wir lange nicht mehr vergessen werden. Wir haben viel gelernt, geflucht, getüftelt, nach Lösungen gesucht und können dennoch mit einem lachenden Auge zurückschauen. Wir könnten sehr viel für unsere beruflichen Laufbahnen mitnehmen und das ist doch das was zählt im Studium.

Wir wollen ein grosses Dankeschön aussprechen an unsere Sponsoren Faulhaber, Büchelblech und die NTB die die Realisierung erst ermöglichten.